

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul efectiv de lucru este de 3 ore.
- În rezolvările cerute, identificatorii utilizați trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată).

(30 Punkte)

Für Punkt 1 schreibt auf das Prüfungsblatt, den Buchstaben welcher der richtigen Antwort entspricht.

```
if (x+y>6) if (x-y<3) cout<<1; | printf("1");
else cout<<2; | printf("2");
else cout<<3; | printf("3");
```

Die ganzen positiven Werte der Variablen x und y für die der Wert 2 angeschrieben wird, sind: (4P.)

d. $x=5$ und $y=3$

```

lies a,b
(natürliche, von Null verschiedene Zahlen,  $a \leq b$ )
i ← a
j ← b
s ← 0
solange i ≤ j wiederhole
| s ← s + (i%2) * i + (j%2) * j
| i ← i + 1
| j ← j - 1
■
schreibe s

```

d. Schreibt das dem gegebenen Algorithmus entsprechende c/c++ Programm. **(10P.)**

(30 Punkte)

Für jeden der Punkte 1 und 2 schreibt auf das Prüfungsblatt, den Buchstaben welcher der richtigen Antwort entspricht.

- ```
struct s1{
 int c,d;
};
struct s2 {
 int a;
 char b;
 struct s1 x;
} y;
```

- a. 5      b. 6      c. 9      d. 10

3. In der nebenstehenden Programmsequenz speichert die Variable **a** ein bidimensionales Feld mit 10 Reihen und 10 Spalten, nummeriert von 1 bis 10, mit den Elementen ganze Zahlen und alle anderen Variablen sind ganz.

```
for(i=1;i<=10;i++)
 for(j=1;j<=10;j++)
 a[i][j]=(i%2)*i+(j%2)*j;
```

4. Sei der gerichtete Graph mit 5 Spitzen, dargestellt durch die nebenstehende Adjazenzmatrix. Schreibt die Menge der Spitzen eines Untergraphs, der eine maximale Anzahl von isolierten Spitzen hat. (6P.)

|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 |

5. Schreibt ein C/C++ Programm, das von der Tastatur einen Text von höchstens 20 Zeichen (kleine und große Buchstaben des englischen Alphabetes) einliest, im Speicher baut und auf dem Bildschirm ausgibt, eine Zeichenfolge gebildet aus den Vokalen die **NICHT** in der eingelesenen Folge erscheinen. Die erwähnten Vokale erscheinen in der erhaltenen Folge in jedwelcher Reihenfolge. Wenn die von der Tastatur eingelesene Folge, alle Vokale enthält, schreibt das Programm die Nachricht **SIRUL VID**. Man nimmt als Vokale an die Buchstaben **a, e, i, o, u, A, E, I, O, U**.

**Beispiel:** wenn man die Folge fulminanta einliest, kann eine Lösung die Folge eioeou sein. (10P.)

**THEMA III**

**(30 Puncte)**

Für Punkt 1 schreibt auf das Prüfungsblatt, den Buchstaben welcher der richtigen Antwort entspricht.

1. Man erzeugt durch die Backtracking Methode alle Werte von natürlichen Zahlen gebildet mit Ziffern der Zahl 356 und die streng kleiner als diese sind. Die ersten fünf erzeugten Zahlen, in dieser Reihenfolge, sind 3, 33, 333, 335, 336. Schreibt die wievielte erzeugte Zahl 5 ist. (4P.)

a. 7                      b. 9                      c. 10                      d. 15

Schreibt auf das Prüfungsblatt die Antwort für jede der folgenden Anforderungen.

2. Sei das Unterprogramm `f`, nebenstehend definierte. Schreibt was für Werte `f(0)`, bzw. `f(14)` haben. (6P.)

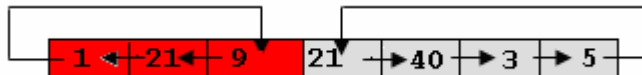
```
int f (int n){
 if (n!=0)
 return n%2+f(n/2);
 else return 0;
}
```

3. Das Unterprogramm `schimb` hat drei Parameter:

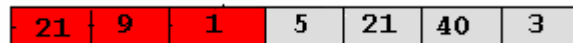
- `n`, durch den es eine natürliche Zahl geliefert bekommt ( $4 < n < 100$ );
- `v`, durch den es ein eindimensionales Feld bekommt, das eine Folge von `n` natürlichen Zahlen speichert, jede mit höchstens vier Ziffern;
- `k`, eine natürliche Zahl ( $1 < k < n-1$ ).

Das Unterprogramm versetzt die ersten `k` Elemente des Feldes kreisförmig nach links, mit je einer Position und die restlichen Elemente kreisförmig nach rechts, mit je einer Position, wie im Beispiel. Das veränderte Feld wird immer durch den Parameter `v` geliefert.

**Beispiel:** wenn `n=7`, `k=3` und das Feld `v` folgende Elemente hat:



nach dem Aufruf ist das Feld:



Schreibt in C/C++ Sprache die vollständige Definition des Unterprogramms. (10P.)

4. Die Textdatei `bac.txt` enthält eine Folge von mindestens drei und höchstens 100000000 ganze Zahlen, jede von höchstens 8 Ziffern. Die Zahlen sind getrennt durch je ein Leerzeichen.

Man nimmt an das eine Folge gebildet aus wenigstens drei Bestandteilen eine arithmetische Folge bildet, wenn die Differenz zwischen jedwelchen Bestandteilen, die sich auf aufeinanderfolgenden Positionen in der Folge befinden, konstant ist.

Man soll die Zahlen aus der Datei `bac.txt` lesen und man soll auf dem Bildschirm die maximale Anzahl der Bestandteile einer Sequenz aus der Folge anschreiben, Sequenz die eine arithmetische Folge bildet. Wenn es keine solche Sequenz gibt, wird auf dem Bildschirm die Nachricht `NU EXISTA` angeschrieben.

Man benützt einen im Bezug auf die Laufzeit und den benötigten Speicher effizienten Algorithmus.

**Beispiel:** wenn die Datei die Zahlen

3 8 13 18 17 16 2 3 9 4 -1 -6 -11 9

enthält, wird auf dem Bildschirm der Wert 5 angeschrieben (entsprechend der Sequenz 9 4 -1 -6 -11).

a) Beschreibt in der Umgangssprache den benötigten Algorithmus und erklärt worin seine Effizienz besteht. (4P.)

b) Schreibt das dem gegebenen Algorithmus entsprechende C/C++ Programm. (6P.)